

UNIVERSITY OF CALIFORNIA

IRVINE

Demonstration of Stroboscopic Algorithm
for Non-Contact Characterization
of Dynamic MEMS

Jason Choi

Harvey Mudd College

Engineering

Mentor:

Professor Andrei M. Shkel

Adam Schofield

Alex Trusov

Ozan Anac

DEDICATION

For
my Family

Contents

Contents	iii
Acknowledgments	v
Abstract	vi
1 Introduction	1
1.1 Motivation	1
1.2 Outline	2
2 Non-Contact Characterization	3
2.1 MEMS Characterization	3
2.2 Difficulties in MEMS Characterization	4
3 Microvision: Stroboscopic Algorithm	6
3.1 Theory	6
3.2 Setup	6
3.2.1 MEMS Device and Actuation	8
3.2.2 Stroboscope Control	9
3.2.3 Calibration	9

3.2.4	Image System Control	10
3.2.5	Data Aquisition	10
4	Experiments and Results	11
5	Summary and Future Work	13
5.1	Summary	13
5.2	Future Work	13
A	ImagePro Macro	17
B	Importing L-edit File into Comsol Multiphysics	22
C	Microvision User Manual	23
C.1	Controlling the Stroboscope	24
C.2	Capturing Images using Image Pro	24
C.3	Setting the Calibration on Image Pro	25
C.4	Image Pro Macro	26
C.5	Matlab M-File	27

Acknowledgments

The success of this project would not have been possible without all the help I have recieved from numerous people through the program.

First, I would like to thank the National Science Foundation and Said Shokair, the Director of the IM-SURE Program, for making this program possible and giving me this great opportunity.

I would like to thank Professor Andrei Shkel for creating this great project and supporting me throughout the entire program. Without his guidelines and words of wisdom, I would not have been able to complete the project.

I am also grateful to Adam Schofield for helping me with all the software and hardware issues relating to the project.

I thank Alex Trusov for helping me learn about MEMS devices and supplying me with devices even though they were important to his own project.

In addition, I thank Ozan Anac for always being there for me to guide me and to support me whenever it was needed.

I am proud of being part of the UCI Microsystems Laboratory and would like to thank all the other graduate students in our lab.

Lastly, I would like to thank all the IM-SURE fellows for giving me a great summer experience.

ABSTRACT

Demonstration of Stroboscopic Algorithm for Non-Contact Characterization of Dynamic MEMS

by

Jason Choi

Mechanical and Aerospace Engineering

University of California, Irvine, 2007

Professor Andrei M. Shkel

Recent development in MEMS tuning fork gyroscopes has created a need for a non-contact characterization method in the UCI Micro Systems Laboratory. Thus, we undertook the task to study and develop a computer micro-vision technique to characterize dynamic MEMS devices. In particular, we developed a system with a stroboscopic algorithm for in-plane dynamics. This approach was based on the use of a strobe light and an image processing software, Image Pro 6.2, to extract the data of the mechanical response of the device. Using the developed micro-vision technique, we were able to identify the dynamic characteristics of several multi-degree of freedom resonators.

Key Terms: MEMS, In-Plane, Microvision, Stroboscopic, Characterization

Chapter 1

Introduction

1.1 Motivation

There has been a rapid development in MEMS technology today; many of such devices use deflection and vibration techniques to achieve measurements of accelerometers or gyroscopes. Thus, there is a demand for a development of techniques to accurately measure and calibrate such MEMS devices. In this project, we tested our MEMS devices using a verified method of video microscopy with a stroboscopic illumination [10]. More specifically, we looked at a gyroscope that uses a tuning fork design. The tuning fork gyroscopes are excited using a known DC and AC voltage in the drive mode. In more detail, as the tuning fork vibrates in the x-axis in the drive mode, if there is a rotation in the z-axis, the coriolis forces make the tuning fork vibrate diagonally with a y-axis component of movement which is the sense mode. This y-axis component of movement can be translated into the angular motion of the gyroscope. To be able to use the tuning fork gyroscopes, we must accurately know the deflection of the device in the drive mode and in this project we measured the deflection using the microvision technique.

One thing that is worthy of mentioning is that Polytec has a vibrometer, namely the MSA-400 that has the capabilities to achieve our goals but to add customizability to the setup,

our lab will produce our own device [11].

1.2 Outline

In this paper we will look in to a method of characterizing MEMS devices, namely the microvision technique that uses the stroboscopic algorithm. This paper includes three parts.

In the first part we will go in depth about MEMS characterization and in the second part we will discuss the design, construction and implementation of the microvision system. In the end, we will conclude the paper with experimental results and future works.

Chapter 2

Non-Contact Characterization

2.1 MEMS Characterization

Today, there are many MEMS devices in development and this has urged today's researchers to develop different characterization tools for MEMS devices. The most common method of characterizing MEMS devices by measuring the capacitance of a moving and a stationary part of a device [1],[2],[3]. These MEMS devices operate using the principles of electrostatics where the stationary part and the part that is allowed to move has been connected to two electrical voltages. By driving the device with a DC coupled with an AC voltage, the part that is allowed to move vibrates at the frequency of the AC voltage and the capacitance can be measured to characterize the system. The device vibration at different amplitudes depending at different frequencies due to the structure of the device and will vibrate at the highest amplitudes at the resonant frequency of the devices. The conventional characterization technique of measuring capacitance however, has drawbacks. The changes in capacitances are small, making measurements hard to measure and inaccurate. Also, this method does not accurately give a physical deflection of the device because the capacitance measurements are translated in to physical deflection by theoretical equations. Thus, in this project, we made a microvision system where we try to maximize the accuracy of the measurement of

the physical deflection of the device. The characterization using microvision will work in two simple steps. First, is a recording step where the actuation of the device is recorded to a video. And the second step is where the frames of the video is analyzed to output a vibration deflection measurement of the device.

2.2 Difficulties in MEMS Characterization

There are three main relevant difficulties in characterizing MEMS device for our project. First, the devices are very small and will have feature sizes in microns. Second, the vibration of the device is very high in the kHz range. And lastly, the vibration amplitudes are very small in the few microns range. All these together make it very hard for a characterization system to measure an amplitude of vibration to high precision. More specifically for our

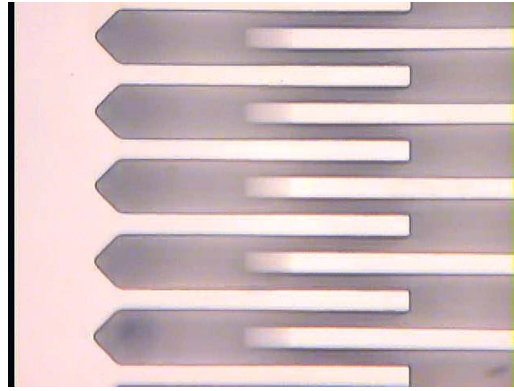


Figure 2.1: Blurred image due to the high frequency vibration of device and low frames per second of video. Photo taken by Alex Trusov.

microvision system, there are difficulties in capturing normal video of the gyroscope due to the camera having a maximum of 60 frames per second whereas the device is vibrating in the KHz rage. To get useful none blurry images of the gyroscope vibrations we use a stroboscope technique. The stroboscope technique uses high frequency flashing light to aid the camera capture a usable picture of the vibration. To get a clear image, the frequency of

the vibrating device must be an integer multiple of the frequency of the flashing light. To overcome the small feature sizes and small deflections we will be using a microscope with the camera to make small measurements.

Chapter 3

Microvision: Stroboscopic Algorithm

3.1 Theory

The microvision method of extracting characterization information of MEMS uses a simple two step process. First, we record the vibration of the device. And second, we extract information such as the amplitude of vibration from the captured frames. There are several proven methods to implement a microvision system, however, in this work we will use a stroboscopic algorithm. The stroboscopic algorithm illuminates the vibrating device to make it appear stationary with a precise frequency [4],[5],[6],[9]. The frequency of vibration of the device must be an integer multiple of the frequency of the stroboscopic illumination.

3.2 Setup

In the following lines I will describe the computer vision system that we have implemented to measure the motion of MEMS devices. We use a microscope to magnify the image, a stroboscopic light to flash the device at a certain frequency and a computer with some software to treat the images [4],[7],[8]. The goal is to obtain non-blur images and be able to play them at a speed that is visible for the human eye. Then by using the computer vision algorithm, we extract the information that we want, the displacement of the moving part.

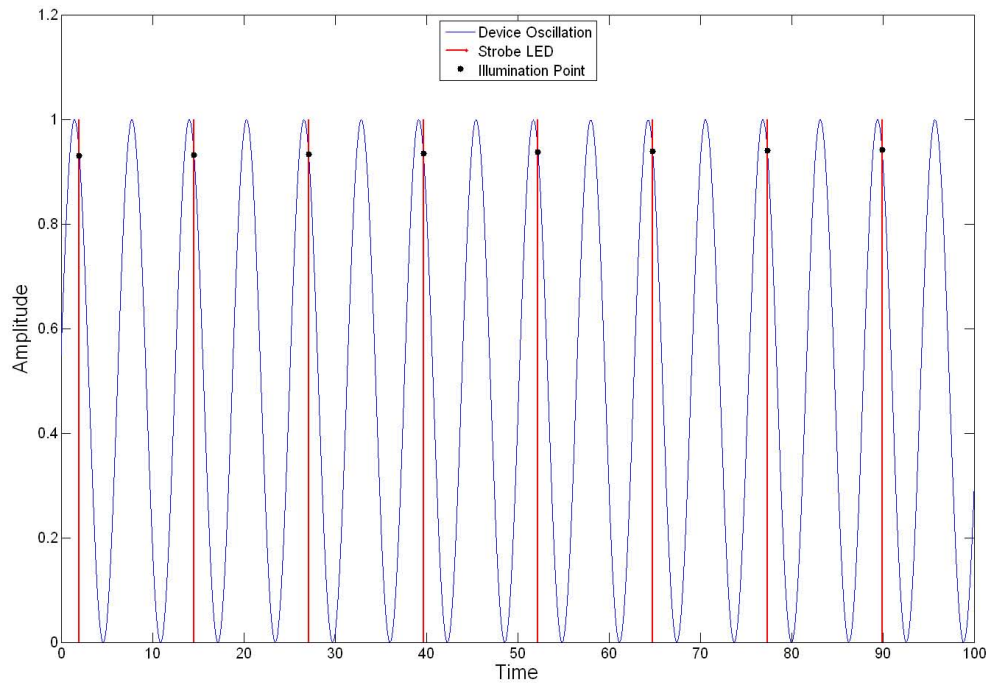


Figure 3.1: Schematic of how the stroboscopic illumination works. If the device oscillates in sinusoid and a stroboscope illumination such as an LED flashes at a certain frequency, only certain portions of the movement will illuminate and make the device appear stationary.

The basic setup of the Microvision system has the following.

- MEMS Device
- Microscope Station
- CCD Camera
- AC Source DC Source Actuating MEMS Device
- Strobe Light Illuminating MEMS Device
- Computer Captures and Analyzes Image from Camera

Here, the AC, DC Sources and Strobe Frequency is controlled manually. However, we created an optional LabVIEW VI that can be used with the microvision system that will automate this process.

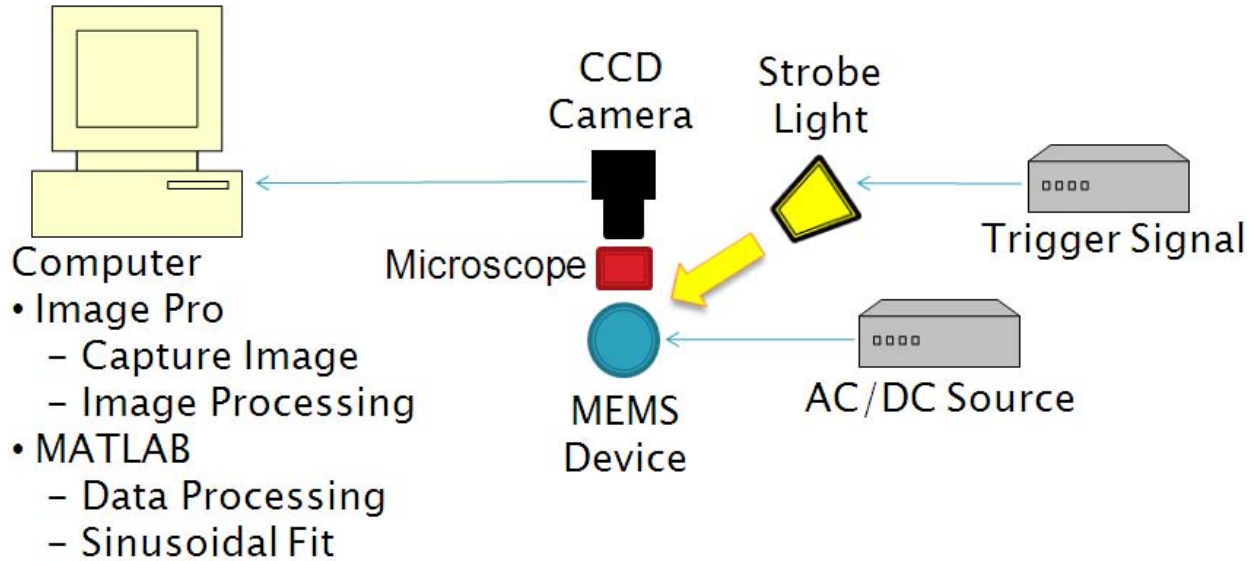


Figure 3.2: Block Diagram of the Microvision System.

3.2.1 MEMS Device and Actuation

For our test, these devices consisted of resonators, which are basic building blocks of tuning fork gyroscopes. It is excited in one axis and under rotation it deflects in another axis that relates to the angular motion. To actuate a resonator, we must supply a DC voltage across the stationary and movable part of the resonator. With a AC voltage coupled with the DC, the resonator starts vibrating in a single axis also known as the drive axis. Due to the mechanical properties of each resonators, each resonator will have a natural frequency of vibration where there is a maximum deflection at a certain frequency. The microvision system will be identifying the deflection at this resonant frequency.

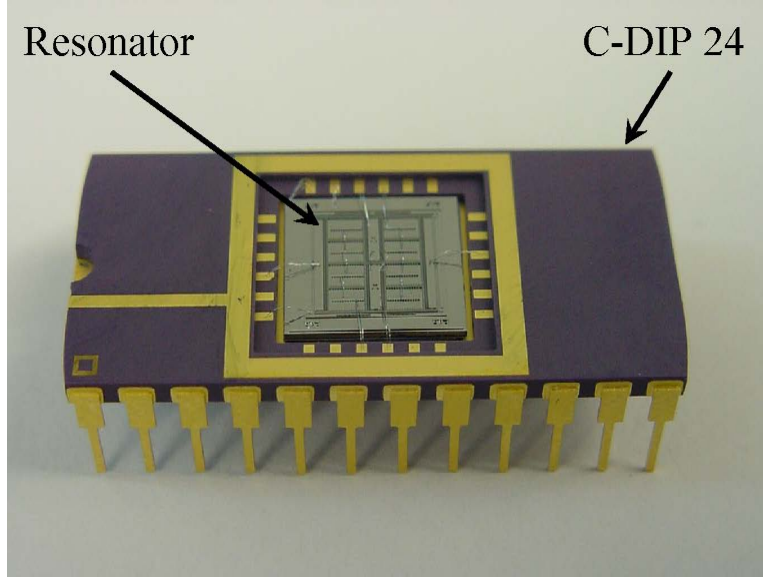


Figure 3.3: Packaged MEMS Device. Fabricated and Photo by Alex Trusov.

3.2.2 Stroboscope Control

The Stroboscope is a light source which flashes light at a given frequency. In this project we used a Perkin Elmer X-Strobe X400. This strobe has a maximum of 60 Hz of flash rate. The flash can be controlled using a 5V TTL pulse where the strobe flashes on the rising edge of the pulse.

3.2.3 Calibration

An image was taken for each of the objectives on the microscope. These pictures were of a calibration grating that had measurements of 10 microns. For experiments using the different objectives, one must change the calibration settings of Image Pro to get the accurate readings of the amplitude of motion.

3.2.4 Image System Control

The camera starts recording a video once it is triggered. The Image Processing software is Image Pro 6.2 and the image acquisition is integrated into the Image Pro Macro.

3.2.5 Data Aquisition

Once an appropriate video is captures from the camera, the video is processed through Image Pro. All a user needs to specify is the line of which the vibration will be analyzed.

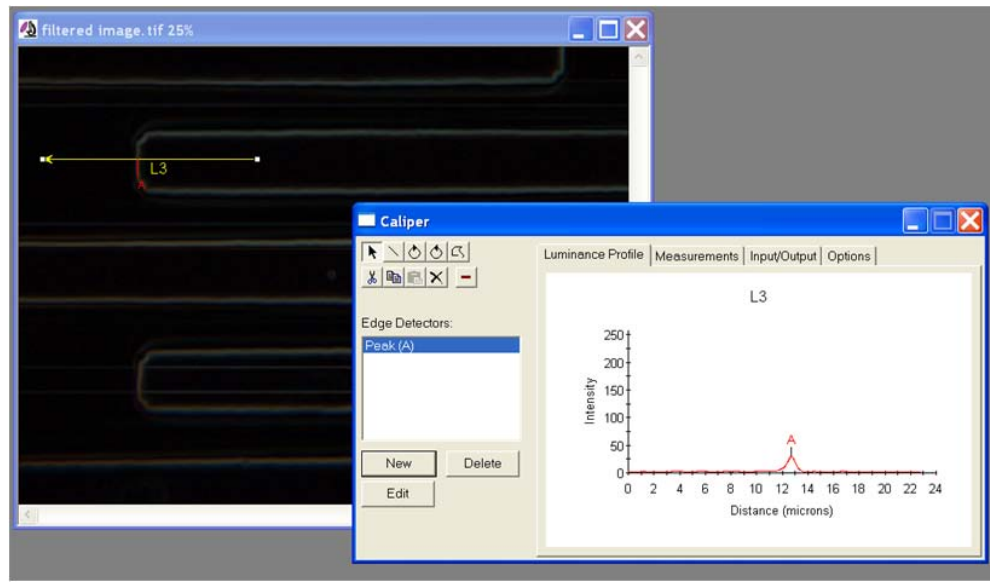


Figure 3.4: Image Pro processing an active image. Here we can see that the Image Pro is identifying the edge of the moving part of the MEMS device. The peak on the graph shows up as A and it identifies the edge.

Chapter 4

Experiments and Results

From Jasmina Casals Thesis, we can see that the data acquired from her microvision system was not perfectly sinusoidal. To determine the amplitude of motion from this data would be difficult and the amplitude would be inaccurate.

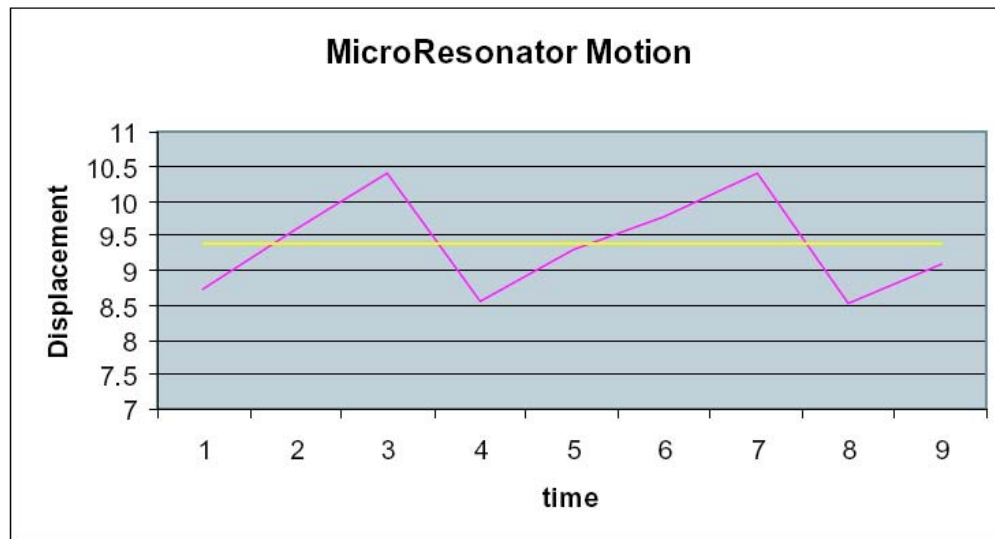


Figure 4.1: Jasmina Casals data from her Thesis, 2002 [10].

Figure 4.2 shows one of the results acquired from the Microvision system that was developed in this project. We can see that a sinusoidal fit is easily fit onto the graph. The amplitude of motion can be found by looking at the amplitude of the sinusoid. To be more specific, the data was from a device that was actuated at 2460 Hz with a frequency of the stroboscope

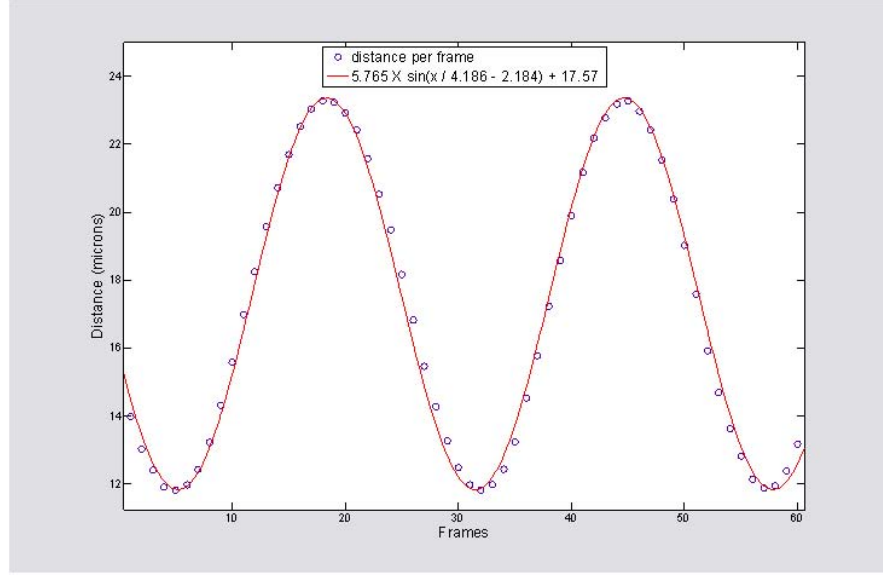


Figure 4.2: Blurred Image.

at 55.909 Hz where integer value $N = 44$. On the graph we can see that the amplitude of motion is 5.77 microns. The uncertainty of this amplitude was also found by taking the standard deviation of the position of a non-moving object. This uncertainty was found to be 0.05 microns. In Figure 4.2 each frame equals to 0.5 seconds. Using this information, we can also verify the frequency vibration of the device, however, the amplitude of motion is the only data we want to extract.

Chapter 5

Summary and Future Work

5.1 Summary

In the following project, the Microvision System was developed and was tested. With the Microvision System working, we were able to measure the amplitudes of vibration of multiple resonators. To make our Microvision System, we made the following tools,

- Image Pro Macro for image capture and image processing.
- Matlab M-File for data processing.
- Optional LabVIEW VI to automatically actuate device and set strobe frequency.
- Successfully camera upgrade.

The uncertainty of our Microvisions system was found to be 0.05 microns.

5.2 Future Work

The future work should include the following,

- Verify the accuracy of the vibration amplitudes with electrical capacitance measurements.

- Characterize more devices.
- Continue Microvision project to characterize 3D movement.

Bibliography

- [1] Stephen D. Senuria, "Microsystem Design.", *Kluwer Academic Publishers*, Boston, 2001.
- [2] Fujita H., Omodaka A. 'The Fabrication of an electrostatic linear actuator by silicon micromachining' *IEEE Trans. on Electron. Dev.* 35, 731-734 (1988)
- [3] Tang W.C., Nguyen T. - C.H., Howe R.T. 'Laterally driven polysilicon resonant microstructures' *Sensors and Actuators* 20, 25-32 (1989)
- [4] D. M. Freeman and C. Q. Davis. Using Video Microscopy to Characterize Micromechanics of Biological and Man-Made Micromechanics. *Technical Digest of the Solid-State Sensor and Actuator Workshop*, June, 1996.
- [5] Matthew R. Hart, Robert A Conant, Kam Y. Lau and Richard S. Muller, Life Fellow, IEEE. Stroboscopic Interferometer System for Dynamic MEMS Characterization. *Journal of Microelectromechanical Systems*, Vol 9, No 4, December, 2000.
- [6] C. Q. Davis and D. M. Freeman. Statistics of subpixel registration algorithms based on spatio-temporal gradients or block matching. *Optical Engineering*, April, 1998.
- [7] Dennis M. Freeman, Alexander J. Aranyosi, Michael J Gordon, Stanley S. Hong. Multi-dimensional Motion Analysis of MEMS using computer Microvision, *Solid State Sensor and Actuator Workshop*, June 8-11, 1998.

- [8] W. Hemmert, M. S. Mermelstein, and D. M. Freeman. Nanometer resolution of three-dimensional motions using video interference microscopy., *MEMS99* Orlando, Fl, 1999.
- [9] Daniel J. Burns and Herbert F. Helbig. A system for Automatic Electrical and Optical Characterization of Microelectromechanical Devies. *Journal of Microelectromechanical Systems*, Vol 8, No. 4, December, 1999.
- [10] Jasmina Casals. "Computer Microvision for In-Plane Non-Contact Characterization of MEMS", *University of California Irvine*, 2002, Thesis.
- [11] Polytec Laser Vibrometry System <http://www.polytec.com/usa/default.asp>

Appendix A

ImagePro Macro

Sub microvision2()

'make sure to set calibration (there are two places in the macros to change this)

'make sure to set how many pictures to take (there are two places in the macros to change this)

'acquire new image

ret = IpAcqSnap(ACQ_NEW)

'select new image

ret = IpAppSelectDoc(1)

'run low pass filter (Kernal size 7, Strength 10, 5 Times)

ret = IpFltLoPass(7,10,5)

'run sobel filter

ret = IpFltSobel()

'close filter window

ret = IpFltShow(0)

'set calibration to N Blue 1X STAR

```

ret = IpSCalSelect("N BLUE 1X STAR")

'open caliper window

ret = IpClprShow(1)

'create edge detectors

ret = IpClprCreateDerivativeEdge("Peak", "A", 255, 0, CLPR_PEAK)

'opens template mode (user may use standard Image-Pro functions)

ret = IpTemplateMode(1)

'pauses macro so the user may choose profile line

ret = IpMacroStop("Choose the Profile Line between moving and fixed parts", 0)

'creates line coordinates

ret = IpListPts(Pts(0), "148 152 235 158")

'creates sampler across the profile line using a caliper

ret = IpClprCreateSampler(CLPR_LINE, "L1", Pts(0), 2)

'close template mode

ret = IpTemplateMode(0)

'set precision

ret = IpClprSet(CLPRSET_PRECISION, 6)

'delete any previous measurement

ret = IpClprDeleteMeas(-1, "", "")

'create new measurement

ret = IpClprCreateMeas(CLPR_MEAS_DIST, "Peak", "")

'save settings such as profile line

ret = IpClprSettings("C:\Documents and Settings\DSP\Desktop\setting.txt", 1)

```

'close caliper tool box

ret = IpClprShow(0)

'close the image

ret = IpDocClose()

'acquire 5 images with .5 second intervals

'the number of images should be equal to seqlong (which controls how many loops will happen)

ret = IpAcqTimed("", "\\NEW\\", 0, 5, .5)

'create integer seqlong

Dim seqlong As Integer

'create integer a

Dim a As Integer

'set seqlong = 5

seqlong = 5

'set a = 0

a = 0

'loop through all the images

While Not(a = seqlong+1)

'select image with number a

ret = IpAppSelectDoc(a)

'run lowpass filter

```

ret = IpFltLoPass(7,10,5)

'run sobel filter

ret = IpFltSobel()

'close filter window

ret = IpFltShow(0)

'set calibration settings

ret = IpSCalSelect("N BLUE 1X STAR")

'close caliper window

ret = IpClprShow(1)

'if first picture is being processed

If a = 0 Then

'load settings

ret = IpClprSettings("C:\Documents and Settings\DSP\Desktop\setting.txt",0)

'save first data

ret = IpClprSave("C:\Documents and Settings\DSP\Desktop\distance.txt", S_DATA2 +
S_FILE)

'if the picture is not the first picture

Else

'close previous image

ret = IpDocCloseEx(a-1)

'load settings

ret = IpClprSettings("C:\Documents and Settings\DSP\Desktop\setting.txt",0)

```

```

'append new data

ret = IpClprSave("C:\Documents And Settings\DSP\Desktop\distance.txt",S_DATA2 +
S_FILE + S_APPEND)

End If

'add 1 to a (this will make the loop consecutively go through all pictures)
a = a+1


'once the loops done....

Wend

'close all images

ret = IpDocClose()

'close caliper window

ret = IpClprShow(0)

End Sub

```

Appendix B

Importing L-edit File into Comsol Multiphysics

- Select All and Flatten.
- Fill in all the gaps using rectangle objects.
- Keep the anchors separate but Merge all the free to move pieces.

(L-edit may add in unnecessary slices that cannot be removed. Ignore this for now.)

- Save the file in L-edit as a GDS file.

(Make sure the correct file is being exported.)

- Import the GDS file into Comsol Multiphysics
- Multiphysics will often fill in portions of the device that is meant to be hollow.

(This problem comes from L-edit adding unnecessary slices in the object.)

- To remove these portions, first Split the object.
- Once the object is split, you will be able to delete portions that are filled in.
- The model should be ready to go.

Appendix C

Microvision User Manual

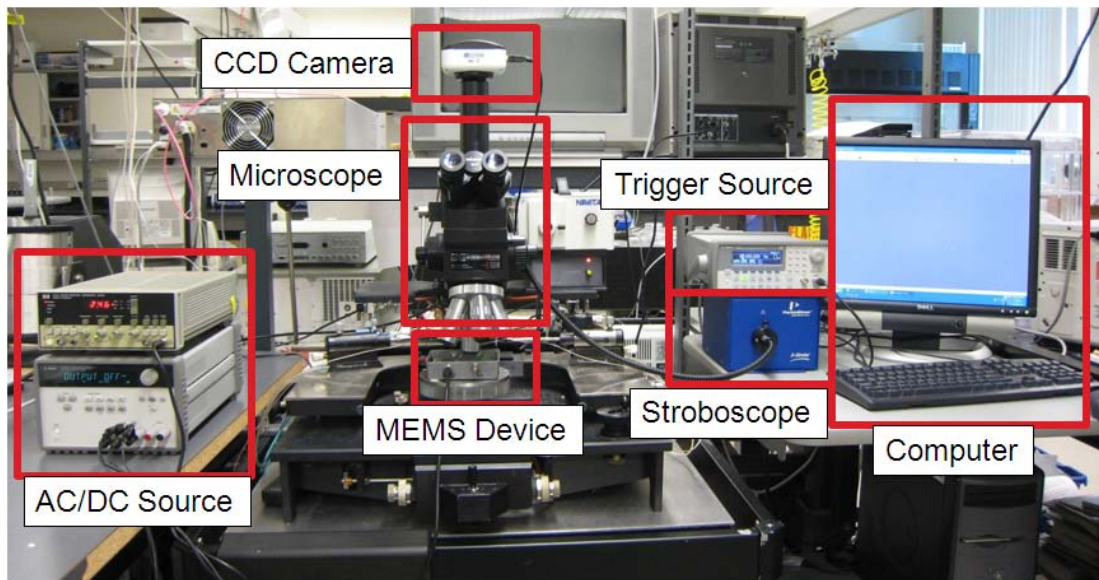


Figure C.1: Microvision System.

First, check for the following items.

- MEMS Device
- AC/DC Source
- Microscope
- CCD Camera connected to computer via USB

- Stroboscope
- Trigger Source
- Computer with ImagePro and Matlab

Next we must make sure that all hardware and software are interfaced correctly.

- Make sure that MEMS device is operational.
- While the MEMS device is in motion, use the trigger source to slow down the motion of the stroboscope.
- Open Image Pro and make sure Image Pro can preview the image from the camera.

Now the Microvision System is ready to capture images and output data.

C.1 Controlling the Stroboscope

There are couple things to remember when controlling the stroboscope. The stroboscope uses a square wave as a trigger signal. The square wave should be a 5 volt square wave less than 60 Hz. To preserve the life of the lamp, please follow the guidelines above.

C.2 Capturing Images using Image Pro

When Image Pro is opened, under the Acquire tab click on Video/Digital Capture. This will open up a window that will allow the user the control the camera settings. Few things the user should know is that the Microsystems Lab has a saved setting "1" that I have found to be optimal for the Microvision macros. However, if there is any problems with the saved settings "1," here are some guidelines to follow. Many of the settings in this menu is divided

into Pvw and Acq. They respectively stand for Preview and Acquire. Thus, the settings that will actually change the Microvision output are the Acquire options. Usually, using the "Auto Set" will put the system at a good setting. However, sometimes Image Pro will tend to overcompensate dark images by increasing the exposure time. If this happens, the user should lower the exposure time, since this will only slow down the Microvision Macro. A good exposure time is around 0.1 to 0.2 seconds. An alternate method to brighten the image is to increase the gain and this will not slow down the acquisition process. The Microvision Macro is set to take in images every 0.5 seconds, thus if the exposure time is longer than this, the Macro will not be able to perform its original purpose. Also, note that there is a resolution settings. After extensive testing, I have found that the computer currently used with the Microvision system cannot handle a constant stream of 5 megapixel images from the camera. Thus, the resolution setting on Image Pro should be either on a lower setting than the highest resolution possible for the camera. Please note that the Microvision Macro will run at the current settings made by the user. Thus, the user should make sure first that the camera settings are good to go before running the Macro. One good test before running the Macro is using the Preview screen to make sure that the moving part of the device is not too blurry. Improving this may be a combination of changing the strobe frequency and camera settings on Image Pro.

C.3 Setting the Calibration on Image Pro

The Calibration process composes of taking a picture of the calibration plate and using Image Pro to set the picture as a calibration. First, use a simple Acquire in the Video/Digital Capture in Image Pro to capture the image of the calibration plate. Save the image with

a proper name and make sure the state which objective was used to capture the image. In Image Pro, open the image. Go to Measure - Calibration - Spatial Calibration Wizard. A new window will show up. Click on calibrate from active image and continue next. Choose microns for the units. You may want to create a new objective and save it. Continue next and you must draw a reference line on the image and state how many microns that line measures. If the image is blurry, you may want to use the optimize button to make a clearer image. Once this is done, the user will be able to see how much each pixel represents in length. One might think this pixel length represents the ultimate limits of the system however, due to vibrations of the microscope and imperfect image processing the actual error in the Microvision system is substantially larger than this value.

There are Calibrations already within the system that I have created

C.4 Image Pro Macro

To load the Macro go to the Macro tab - Macro - Change - Choose the Macro to be loaded and click Open - Push Ok. This will load the Macro into Image Pro. Now to run the selected Macro go to the Macro tab and click on the Macro that has just been loaded. The Macro should be loaded up as the last option in the Macro tab. Once the Macro starts running, there are couple things the user must do for the Macro to run successfully. When the Edge Detector window opens, double click on the Edge Detector Peak(A) and put in as the sensitivity threshold to 3 if it is not already. This threshold value is the threshold for how high the peak needs to be in the edge detection and for different devices this value may need to vary. The goal is to pick a value such that the peak detector finds the edge and no other false edges. I have found that values from 3 to 5 are most successful with the current

image filters in the Microvision Macro. Make sure the choose the profile line, where the edge will be found and check that the edge is being properly detected before continuing on with the Macro.

C.5 Matlab M-File

There are two files created by ImagePro. The settings and the data. Once the Image Pro Macro runs a text file distance.txt will be created on the desktop. Make sure to rename this file, otherwise when the Macro runs again, all the previous data will be lost. To run the Matlab M-File on the distance data, first open Matlab and change the current directory to the Desktop. Double click on the distance.txt file or the renamed file. On the keyboard push Ctrl + F and the find and replace box will pop up. Here, in the find what section type in "A" without the quotation marks and click replace all. The "A" is the distance file is useless information and will interfere with the M-File. Now save the file. Now in the Command Window, import this file into Matlab using the `uiimport('file address')`. The Import Wizard Window will open. Click on Space in the Select Column Separator(s). On the Number of text header lines put 2. Click Next and in the next window click Finish. In the command window type in `process(data,textdata);` and execute the command by pressing enter. A plot should show up with a sinusoidal fit to the points extracted by ImagePro. The coefficient information will show in the Command Window. However, because the sinusoidal fit is not perfect, it will not always create a correct fit. If this happens use `process2(data,textdata);`. This will open a curve fitting tool and one will have to manually create a sinusoidal fit graph.